Abstract

Supervised learning of piecewise linear models is a well studied problem in machine learning community. The key idea in piecewise linear modeling is to properly partition the input space and learn a linear model for every partition. Decision trees and regression trees are classic examples of piecewise linear models for classification and regression problems.

The existing approaches for learning decision/regression trees can be broadly classified into two classes, namely, fixed structure approaches and greedy approaches. In the fixed structure approaches, tree structure is fixed beforehand by fixing the number of non-leaf nodes, height of the tree and paths from root node to every leaf node of the tree. Mixture of experts and hierarchical mixture of experts are examples of fixed structure approaches for learning piecewise linear models. Parameters of the models are found using, e.g., maximum likelihood estimation, for which expectation maximization (EM) algorithm can be used. Fixed structure piecewise linear models can also be learnt using risk minimization under an appropriate loss function. Learning an optimal decision tree using fixed structure approach is a hard problem. Constructing an optimal binary decision tree is known to be NP-Complete. On the other hand, greedy approaches do not assume any parametric form or any fixed structure for the decision tree classifier. Most of the greedy approaches learn tree structured piecewise linear models in a top-down fashion. These are built by binary or multi-way recursive partitioning of the input space. The main issues in top-down decision tree induction is to choose an appropriate objective function to rate the split rules. The objective function should be easy to optimize. Top-down decision trees are easy to implement and understand, but there are no optimality guarantees due to their greedy nature. Regression trees are built in the similar way as decision trees. In regression trees, every leaf node is associated with a linear regression function.

All piecewise linear modeling techniques deal with two main tasks, namely, partitioning of the input space and learning a linear model for every partition. However,

partitioning of the input space and learning linear models for different partitions are not independent problems. Simultaneous optimal estimation of partitions and learning linear models for every partition, is a combinatorial problem and hence computationally hard. However, piecewise linear models provide better insights into the classification or regression problem by giving explicit representation of the structure in the data. The information captured by piecewise linear models can be summarized in terms of simple rules, so that, they can be used to analyze the properties of the domain from which the data originates. These properties make piecewise linear models, like decision trees and regression trees, extremely useful in many data mining applications and place them among top data mining algorithms.

In this thesis, we address the problem of supervised learning of piecewise linear models for classification and regression. We propose novel algorithms for learning piecewise linear classifiers and regression functions. We also address the problem of noise tolerant learning of classifiers in presence of label noise.

We propose a novel algorithm for learning polyhedral classifiers which are the simplest form of piecewise linear classifiers. Polyhedral classifiers are useful when points of positive class fall inside a convex region and all the negative class points are distributed outside the convex region. Then the region of positive class can be well approximated by a simple polyhedral set. The key challenge in optimally learning a fixed structure polyhedral classifier is to identify subproblems, where each subproblem is a linear classification problem. This is a hard problem and identifying polyhedral separability is known to be NP-complete. The goal of any polyhedral learning algorithm is to efficiently handle underlying combinatorial problem while achieving good classification accuracy. Existing methods for learning a fixed structure polyhedral classifier are based on solving non-convex constrained optimization problems. These approaches do not efficiently handle the combinatorial aspect of the problem and are computationally expensive. We propose a method of model based estimation of posterior class probability to learn polyhedral classifiers. We solve an unconstrained optimization problem using a simple two step algorithm (similar to EM algorithm) to find the model parameters. To the best of our knowledge, this is the first attempt to form an unconstrained optimization problem for learning polyhedral classifiers. We then modify our algorithm to find the number of required hyperplanes also automatically. We experimentally show that our approach is better than the existing polyhedral learning algorithms in terms of training time, performance and

the complexity.

Most often, class conditional densities are multi-modal. In such cases, each class region may be represented as a union of polyhedral regions and hence a single polyhedral classifier is not sufficient. To handle such situation, a generic decision tree is required. Learning optimal fixed structure decision tree is a computationally hard problem. On the other hand, top-down decision trees have no optimality guarantees due to the greedy nature. However, top-down decision tree approaches are widely used as they are versatile and easy to implement. Most of the existing top-down decision tree algorithms (CART, OC1, C4.5, etc.) use impurity measures to assess the goodness of hyperplanes at each node of the tree. These measures do not properly capture the geometric structures in the data. We propose a novel decision tree algorithm that, at each node, selects hyperplanes based on an objective function which takes into consideration geometric structure of the class regions. The resulting optimization problem turns out to be a generalized eigen value problem and hence is efficiently solved. We show through empirical studies that our approach leads to smaller size trees and better performance compared to other top-down decision tree approaches. We also provide some theoretical justification for the proposed method of learning decision trees.

Piecewise linear regression is similar to the corresponding classification problem. For example, in regression trees, each leaf node is associated with a linear regression model. Thus the problem is once again that of (simultaneous) estimation of optimal partitions and learning a linear model for each partition. Regression trees, hinge hyperplane method, mixture of experts are some of the approaches to learn continuous piecewise linear regression models. Many of these algorithms are computationally intensive. We present a method of learning piecewise linear regression model which is computationally simple and is capable of learning discontinuous functions as well. The method is based on the idea of K-plane regression that can identify a set of linear models given the training data. K-plane regression is a simple algorithm motivated by the philosophy of k-means clustering. However this simple algorithm has several problems. It does not give a model function so that we can predict the target value for any given input. Also, it is very sensitive to noise. We propose a modified K-plane regression algorithm which can learn continuous as well as discontinuous functions. The proposed algorithm still retains the spirit of k-means algorithm and after every iteration it improves the objective function. The proposed method learns a proper

piecewise linear model that can be used for prediction. The algorithm is also more robust to additive noise than K-plane regression.

While learning classifiers, one normally assumes that the class labels in the training dataset are noise free. However, in many applications like Spam filtering, text classification etc., the training data can be mislabeled due to subjective errors. In such cases, the standard learning algorithms (SVM, Adaboost, decision trees etc.) start overfitting on the noisy points and lead to poor test accuracy. Thus analyzing the vulnerabilities of classifiers to label noise has recently attracted growing interest from the machine learning community. The existing noise tolerant learning approaches first try to identify the noisy points and then learn classifier on remaining points. In this thesis, we address the issue of developing learning algorithms which are inherently noise tolerant. An algorithm is inherently noise tolerant if, the classifier it learns with noisy samples would have the same performance on test data as that learnt from noise free samples. Algorithms having such robustness (under suitable assumption on the noise) are attractive for learning with noisy samples. Here, we consider nonuniform label noise which is a generic noise model. In nonuniform label noise, the probability of the class label for an example being incorrect, is a function of the feature vector of the example. (We assume that this probability is less than 0.5 for all feature vectors.) This can account for most cases of noisy datasets. There is no provably optimal algorithm for learning noise tolerant classifiers in presence of nonuniform label noise. We propose a novel characterization of noise tolerance of an algorithm. We analyze noise tolerance properties of risk minimization framework as risk minimization is a common strategy for classifier learning. We show that risk minimization under 0-1 loss has the best noise tolerance properties. None of the other convex loss functions have such noise tolerance properties. Empirical risk minimization under 0-1 loss is a hard problem as 0-1 loss function is not differentiable. We propose a gradient free stochastic optimization technique to minimize risk under 0-1 loss function for noise tolerant learning of linear classifiers. We show (under some conditions) that the algorithm converges asymptotically to the global minima of the risk under 0-1 loss function. We illustrate the noise tolerance of our algorithm through simulations experiments. We demonstrate the noise tolerance of the algorithm through simulations.